**CHAPTER 3: DESIGN**

**Introduction**

Design (Janeth, 2001) is considered as one of the most vital phase in the process of development of software. It shows that how the system would perform. Here, all the modeling is performed with their respective diagrams such as structural modeling, behavioral modeling, database modeling and UI modeling.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S. No.** | **Structural Modeling** | **Behavioral Modeling** | **Database Modeling** | **UI Modeling** |
| 1 | Final Class Diagram | Activity Diagram | Data Dictionary | Prototype |
| 2 | Flow Chart | Sequence Diagram | ER Diagram |  |

**3.1 Structural Modeling**

**3.1.1 Final Class Diagram**

Class diagram (Hereby, 2002) is one of the types of static diagram which describes the structures of the system showing its classes, objects, attributes, operations and its relationships among objects. The following diagram will make clear. It makes us clear that how the system does work with one another relations.

|  |  |
| --- | --- |
| **S. No.** | **Notation Used** |
| 1 | Class |
| 2 | Directed Association |
| 3 | Operations |

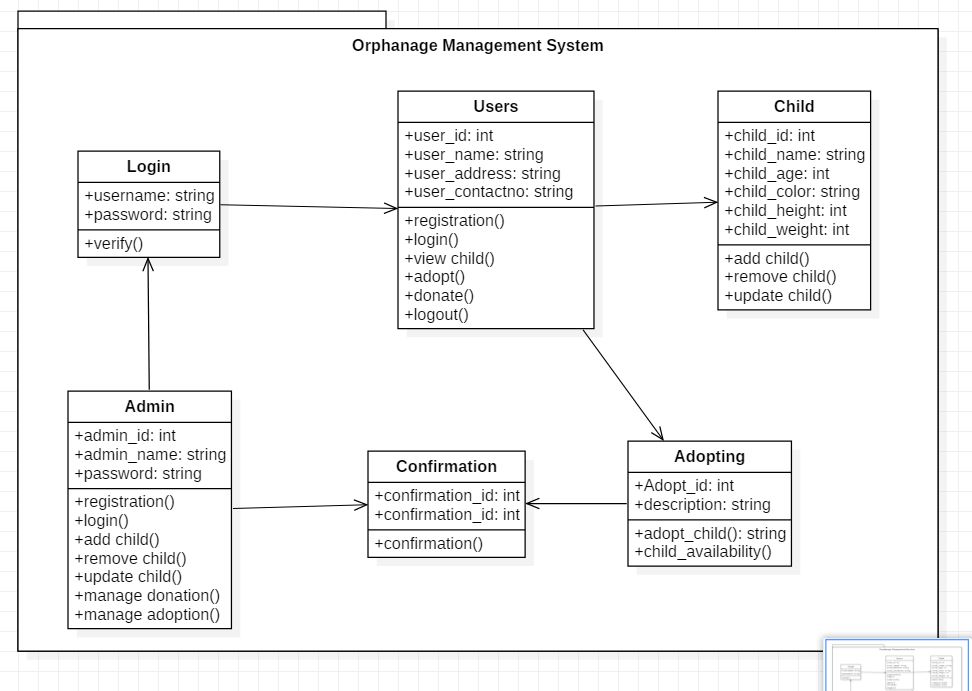


Figure 1: Final Class Diagram

Here, admin as well as users login to the system. Users could make update their profiles and donate some money as well as adopt child. After adopting child, they get a confirmation id in order to clear doubts and be safe from frauds. Similarly, admin make login in order to update the whole system.

**3.1.2 Flow Chart**

A flow chart (Smith, 2000) is a diagram that depicts the process, system and the algorithms. It is also known as Process Flow Chart, Process Map, and Functional Flow Chart and so on. Here, we could see the flow of the system that how it works in practical with the help of diagram.

|  |  |
| --- | --- |
| **S. No.** | **Notation Used** |
| 1 | Flow |
| 2 | Processes |
| 3 | Terminators |
| 4 | Decision |

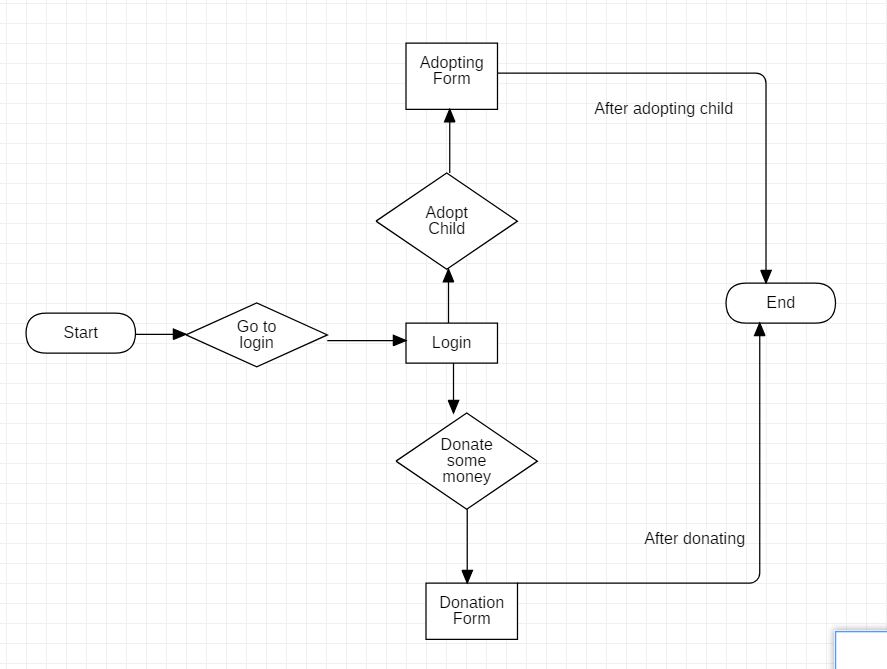


Figure 2: Flow Chart

Firstly, the users go to the login page and make login. Then, they could adopt child as their wish by visiting the adaptation page. Similarly, they could also donate money as per their wishes by visiting the donation page. Finally, after doing all these processes ends.

**3.2 Behavioral Modeling**

**3.2.1 Activity Diagram**

Activity diagrams (Manieepo, 2004) are used to illustrate the flow of control in the system and refers to the steps involved in the execution of the use case diagram. It gives the details information about the working processes of the system. The following diagrams will make all the confusions clear and helps to understand in a good and untestable manner.

|  |  |
| --- | --- |
| **S. No.** | **Notation Used** |
| 1 | Action |
| 2 | Initial |
| 3 | Final |
| 4 | Fork |
| 5 | Join |
| 6 | Initial |
| 7 | End |
| 8 | Decision |
| 9 | Control Flow |

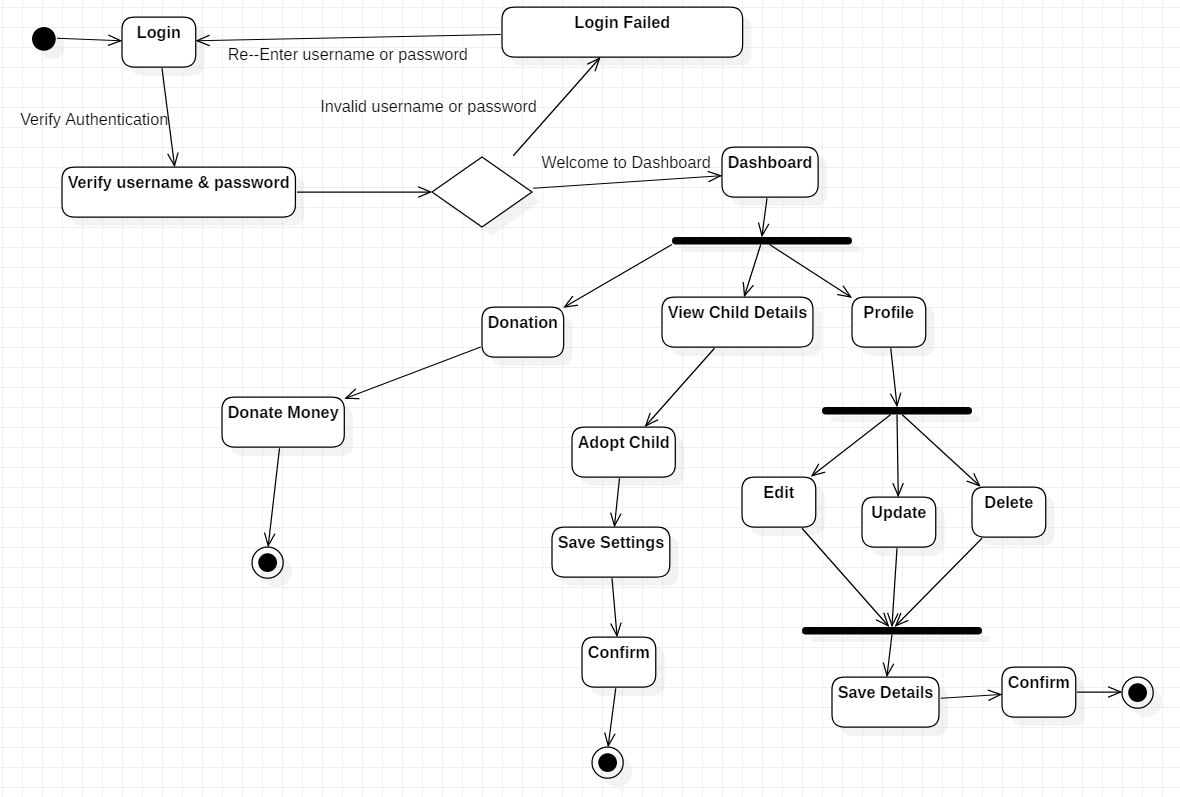


Figure 3: Activity Diagram

First of all, users make login to the system by entering their usernames and passwords. If their usernames and passwords are valid then they are taken to the dashboard. But if their username and password are invalid then they are requested for re-entering their valid usernames and passwords. After visiting the dashboard, they could visit their profile and make some updates, edits and deletes and then save those all details. Similarly, they could also view child details and adopt child. Similarly, they could also donate some money to the orphanage.

**3.2.2 Sequence Diagram**

Sequence diagrams (Beekam, 2001) explain the interactions among all classes in terms of an exchange of messages over time. They're also known as “Event Diagrams”. It is one of the good ways to visualize and validate the various runtime scenarios. These can help us to predict that how the system would behave and to discover responsibilities a class may need to have in the process of modeling a new system.

|  |  |
| --- | --- |
| **S. No.** | **Notation Used** |
| 1 | Lifeline |
| 2 | Message |
| 3 | Reply message |
| 4 | Self-message |

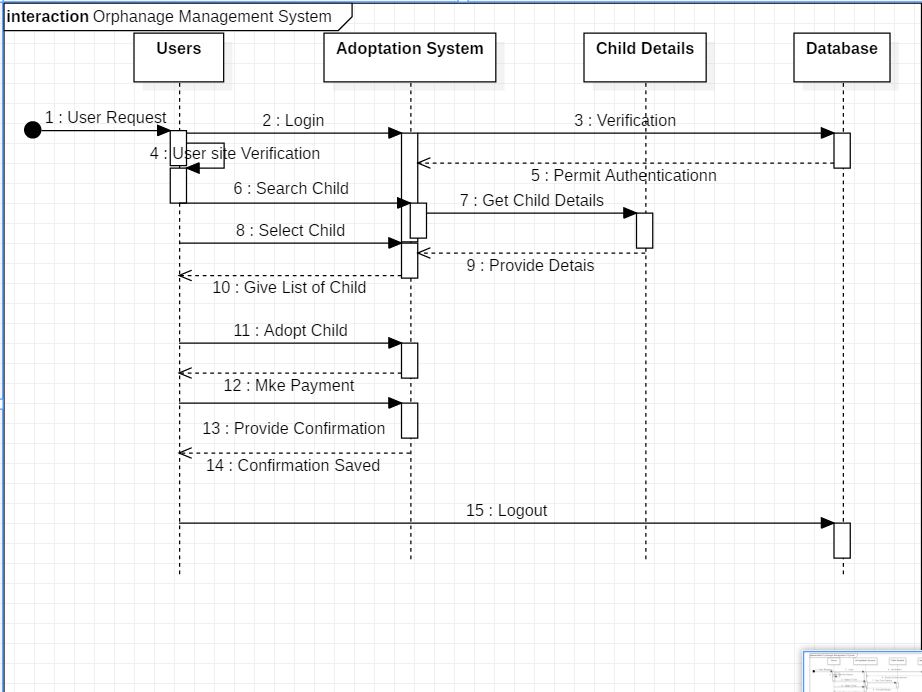


Figure 4: Sequence Diagram

According to the diagram, users make request for login and then their usernames and passwords are verified. After that users could search for the child and get details of child. Then, they could adopt child and make payment. The system asks for the confirmation that users had given for being safe from frauds.

**3.3 Database Modeling**

**3.3.1 Data Dictionary**

A data dictionary is a metadata that contains the data about the data. It could be also called as data about the database. It contains the very confidential data and information about the admin as well as users and the orphanage too. Therefore, it is very important to the system. The following data dictionary would make us clear.

1. **Child Details**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Type** | **size** | **Key** | **Description** |
| child\_id | Integer | 20 | Primary | Id of the child |
| child\_firstname | Varchar | 20 |  | Fistname of the child |
| child\_lastname | Varchar | 20 |  | Lastname of the child |
| child\_height | Varchar | 20 |  | Height of the child. |
| child\_weight | Varchar | 20 |  | Weight of the child. |
| child\_color | Integer | 20 |  | ----------- |
| adopt\_id | varchar | 20 | Foreign | Color of the child. |

1. **Admin Details**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Type** | **size** | **Key** | **Description** |
| admin\_id | Integer | 20 | Primary | Id of the admin |
| admin\_name | Varchar | 20 |  | Name of the admin. |
| admin\_contact | Varchar | 20 |  | Contact of the admin. |
| admin\_address | Varchar | 20 |  | Address of the admin. |

1. **User Details**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Type** | **size** | **Key** | **Description** |
| user\_id | integer | 20 | Primary | Id of the user |
| user\_name | Varchar | 20 |  | Name of the user. |
| user\_contact | Varchar | 20 |  | Contact of the user. |
| user\_address | Varchar | 20 |  | Address of the user. |

1. **Adopt Details**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Type** | **size** | **Key** | **Description** |
| adopt\_id | integer | 20 | Primary | Id of the adopt |
| user\_id | Varchar | 20 | Foreign | Id of the user. |

1. **Confirmation Details**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Type** | **size** | **Key** | **Description** |
| confirmation\_id | Integer | 20 | Primary | Id of th cofirmation. |
| admin\_id | Integer | 20 | Foreign | Id of the admin\_id. |

**3.3.2 Entity Relationship Diagram**

An Entity Relationship Diagram (Anopr, 2002) is one of the visual paradigms that show the relationships of the entities sets stored in the database. The entity set is a collection of similar entities and these can have attributes which defines its properties. We would be clear with the help of the following diagram.

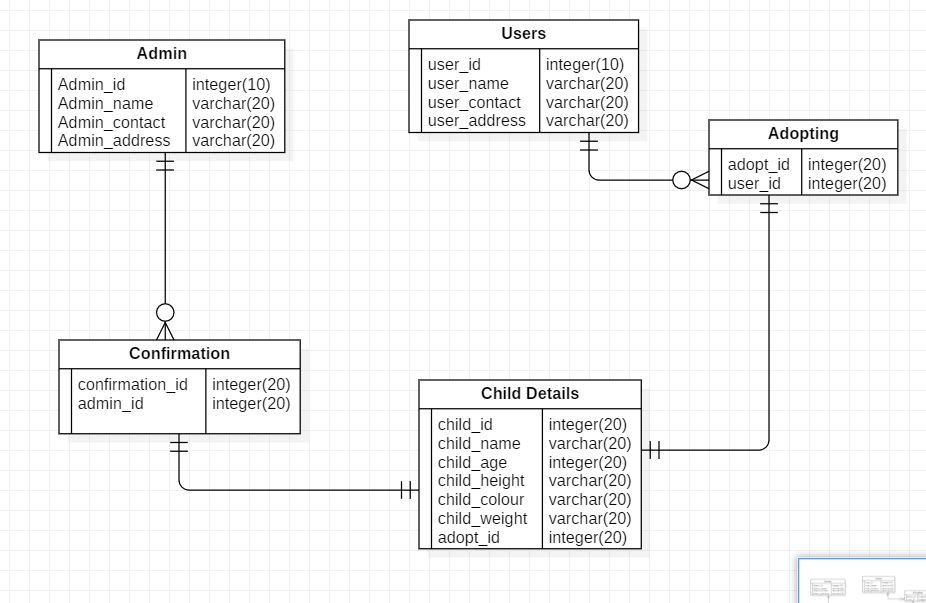


Figure 5: ER-Diagram

As per the diagram, there is many to one relationship between Confirmation and Admin. Similarly, there is one to one relationship between Child Details and Confirmation. Not only that there is also one to one relationship between Child Details and Adopting. Likewise, there is one to many relationship between Users and Adopting.

**3.4 UI Modeling**

**3.4.1 Prototype**

A prototype is like a draft representation that is built to test ideas for layout, behavior and flow in a system. They are indispensable tools for resolving the large numbers of potential issues in a concept or business before too many resources are deployed to put a design into the production.

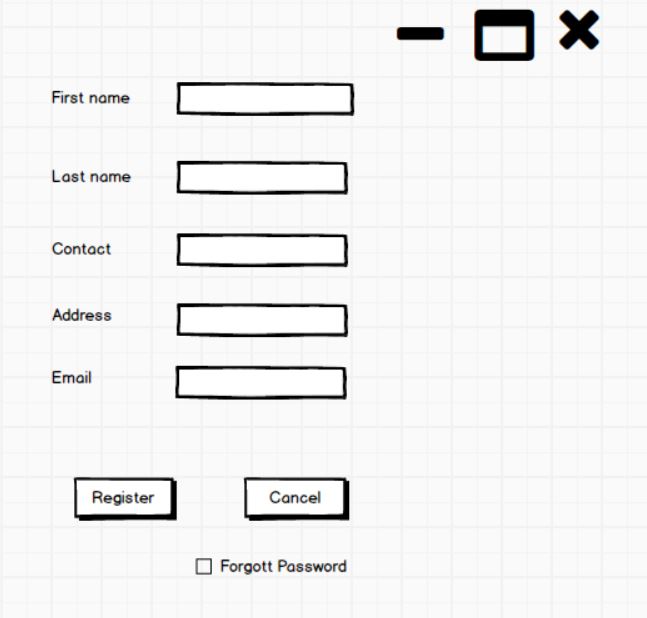


Figure 6: Screenshot of registration form

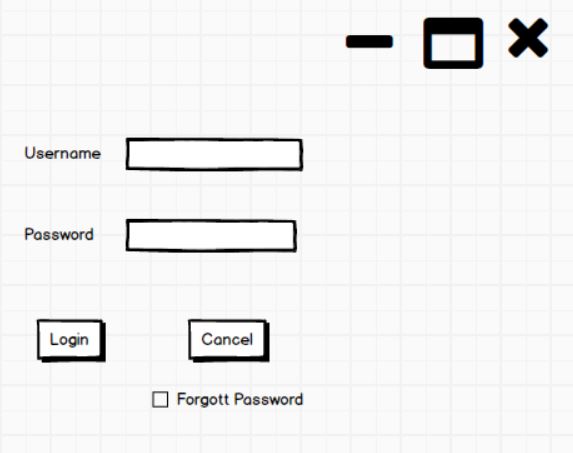


Figure 7: Screenshot of login form

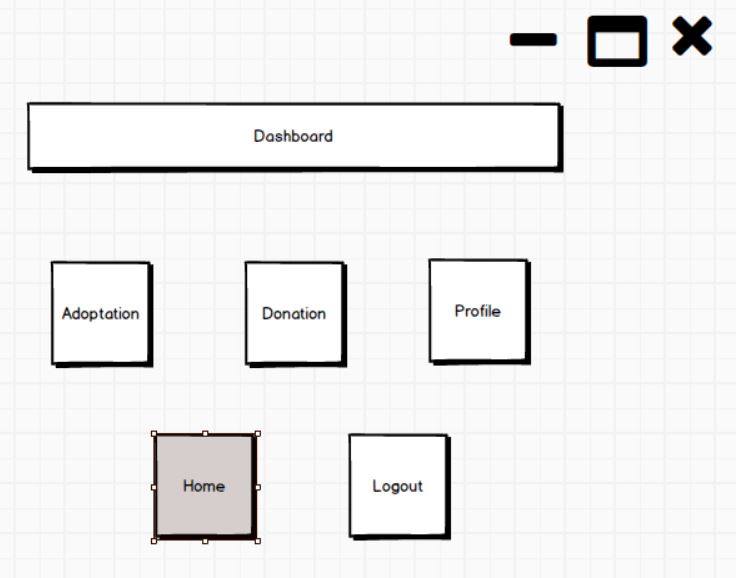


Figure 8: Screenshot of dashboard

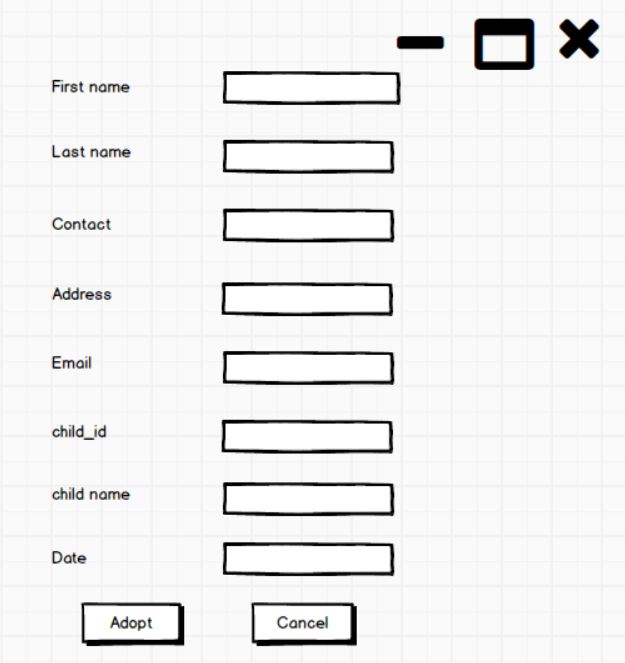


Figure 9: Screenshot of adopting Form

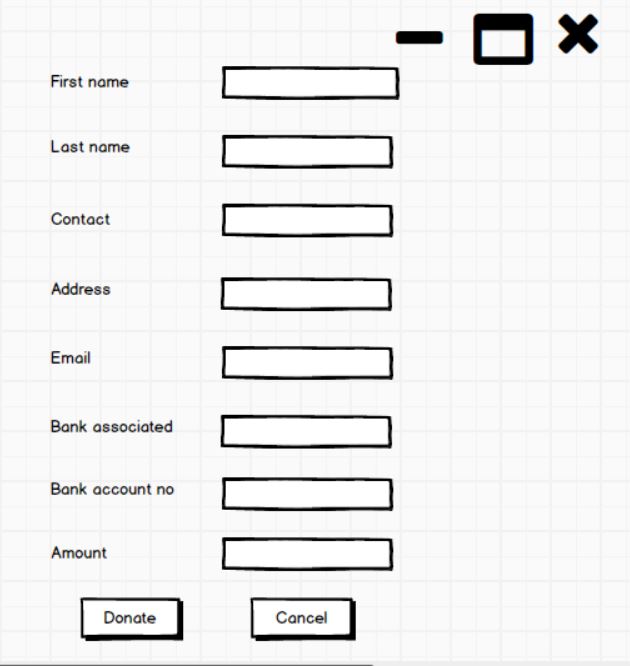


Figure 10: Screenshot of donation form

**Conclusion**

In this way, the system is going to work when it is developed. The above all diagrams and their explanations clarify the working processes of the system one by one. In order to complete these all tasks, some tools are used like Star UML and Balsamic.

# References

* Anopr. (2002). Retrieved from www.tutorialspoint.com: https://www.tutorialspoint.com/What-is-Data-Dictionary
* Beekam. (2001). Retrieved from www.geeksforgeeks.org: https://www.geeksforgeeks.org/unified-modeling-language-uml-sequence-diagrams/
* Hereby. (2002). Retrieved from www.visual-paradigm.com: https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-class-diagram/
* Janeth. (2001). Retrieved from www.quora.com: https://www.quora.com/What-does-program-design-mean-in-programming
* Manieepo. (2004). Retrieved from www.visual-paradigm.com: https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-activity-diagram/
* Smith, W. (2000). Retrieved from www.breezetree.com: https://www.breezetree.com/articles/what-is-a-flow-chart